SIAM Conference on Computational Science and Engineering

Short Course on the ACTS Collection:

Robust and High Performance Libraries for Computational Sciences

The DOE ACTS Collection
- Closing Remarks -

Osni Marques

Lawrence Berkeley National Laboratory (LBNL)

oamarques@lbl.gov

Challenges in the Development of Scientific Codes

- Productivity
  - Time to the first solution (prototype)
  - Time to solution (production)
  - Other requirements
- Complexity
  - Increasingly sophisticated models
  - Interdisciplinarity
  - Model coupling
- Performance
  - Increasingly complex algorithms
  - Increasingly complex architectures
  - Increasingly demanding applications

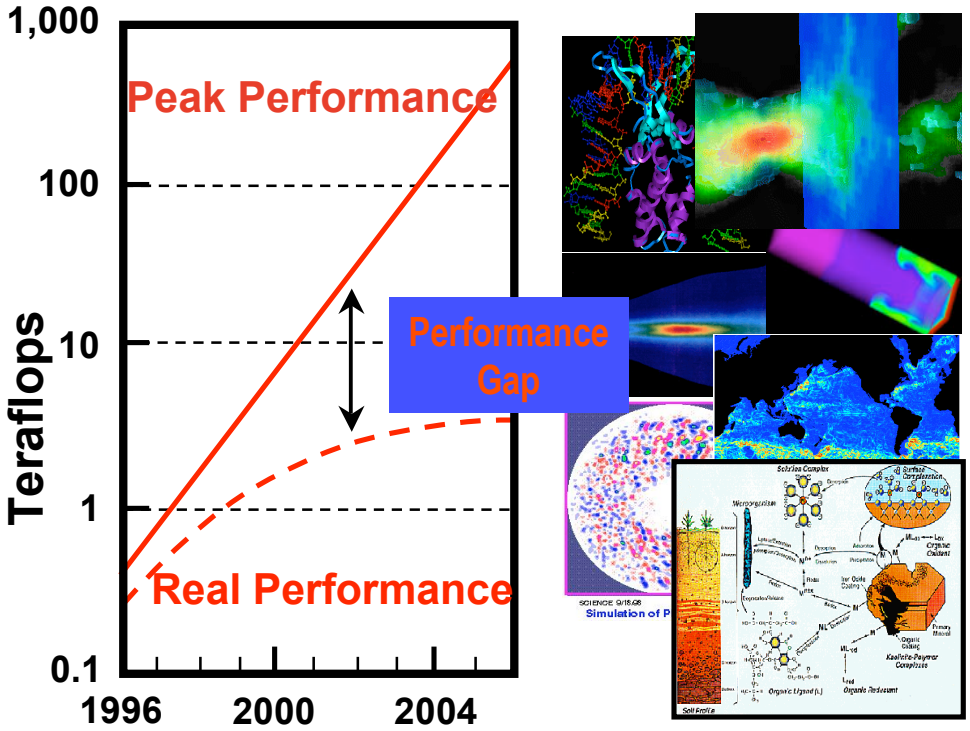
- Libraries written in different languages.
- Different pieces of the code evolve at different rates
- Swapping competing implementations of the same idea and testing without modifying the code

Peak performance is skyrocketing

- *In 1990s, peak performance increased 100x; in 2000s, it will increase 1000x*

However

- *Efficiency for many science applications declined from 40-50% on the vector supercomputers of 1990s to as little as 5-10% on parallel supercomputers of today*



The Reality…

- Research in computational sciences is fundamentally interdisciplinary and addresses, among many others, climate and environment modeling, DNA sequencing, flows in geological structures, etc.
- In 1999, the PITAC Report recommended the creation of a national library of certified domain-specific software in order to reduce the labor required for software development, testing and evolution.
- The development of complex simulation codes on high-end computers is not a trivial task.
- Usually, a significant percentage of the efforts focus on the development of the codes and their optimization.
- There is a need for a collaboration framework for ongoing development and deployment of computational tools.

See SCaLes (Science Case for Large-scale Simulation) and HECRTF (High Computing Revitalization Task Force) reports.

The DOE ACTS Collection

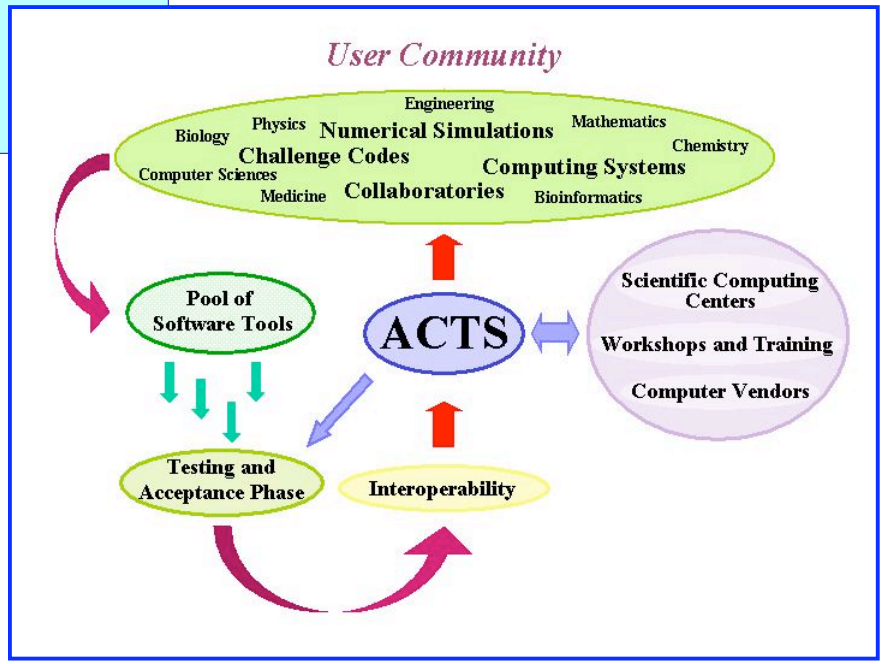
Goals

- *Collection of tools for developing parallel applications*
- *Extended support for experimental software*
- *Make ACTS tools available on DOE computers*
- *Provide technical support (acts-support@nersc.gov)*
- *Maintain ACTS information center (http://acts.nersc.gov)*
- *Coordinate efforts with other supercomputing centers*
- *Enable large scale scientific applications*
- *Educate and train*

Levels of Support

- ***High***
  - Intermediate level
  - Tool expertise
  - Conduct tutorials
- ***Intermediate***
  - Basic level
  - Higher level of support to users of the tool
- ***Basic***
  - Help with installation
  - Basic knowledge of the tools
  - Compilation of user's reports

- High Performance Tools
  - *portable*
  - *library calls*
  - *robust algorithms*
  - *help code optimization*
- More code development in less time
- More simulation in less computer time



Current ACTS Tools and their Functionalities

Category	Tool	Functionalities
Numerical	Aztec/Trilinos	Algorithms for the iterative solution of large sparse linear systems.
	Hypre	Algorithms for the iterative solution of large sparse linear systems, intuitive grid-centric interfaces, and dynamic configuration of parameters.
	PETSc	Tools for the solution of PDEs that require solving large-scale, sparse linear and nonlinear systems of equations.
	OPT++	Object-oriented nonlinear optimization package.
	SUNDIALS	Solvers for the solution of systems of ordinary differential equations, nonlinear algebraic equations, and differential-algebraic equations.
	ScaLAPACK	Library of high performance dense linear algebra routines for distributed-memory message-passing.
	SuperLU	General-purpose library for the direct solution of large, sparse, nonsymmetric systems of linear equations.
	TAO	Large-scale optimization software, including nonlinear least squares, unconstrained minimization, bound constrained optimization, and general nonlinear optimization.
Code Development	Global Arrays	Library for writing parallel programs that use large arrays distributed across processing nodes and that offers a shared-memory view of distributed arrays.
	Overture	Object-Oriented tools for solving computational fluid dynamics and combustion problems in complex geometries.
Code Execution	CUMULVS	Framework that enables programmers to incorporate fault-tolerance, interactive visualization and computational steering into existing parallel programs
	Globus	Services for the creation of computational Grids and tools with which applications can be developed to access the Grid.
	TAU	Set of tools for analyzing the performance of C, C++, Fortran and Java programs.
Library Development	ATLAS	Tools for the automatic generation of optimized numerical software for modern computer architectures and compilers.

Who Benefits from these tools?

Application	Computational Problem	Software Tools	Highlights
MADCAP	Matrix factorization and triangular solves	ScaLAPACK	• 50% peak performance on an IBM SP • Nearly perfect scalability on 1024, 2048, 3072 and 4096 processors • Fast implementation of numerical algorithms
3-Charged Particles	Solution of large, complex unsymmetric linear systems	SuperLU	• Solves systems of equations of order 8.4 million on 64 processors in 1 hour of wall clock time • 30 GFLOPs
NWChem	Distribute large data arrays, collective operations	Global Arrays and LAPACK	• Very good scaling for large problems
FUN3D &	Unstructured grids, compressible and incompressible Euler and Navier-Stokes equations.	PETSc	• Parallelization of legacy code • Gordon Bell price, 0.23 Tflop/s on 3072 procs of ASCI Red
P-FLAPW	Eigenvalue problems	ScaLAPACK	• Study of systems up to 700 atoms (mat size=35,000) • Runs efficiently • Facilitated the study of new problems in materials science such as impurities and disordered systems.
NIMROD	Quad and triangular high order finite elements, semi-implicit time integration,sparse matrix solvers	SuperLU	• Code improvement of 5 fold, equivalent to 3-5 years progress in computing hardware.

http://acts.nersc.gov/AppMat

Enabling sciences and discoveries…

with

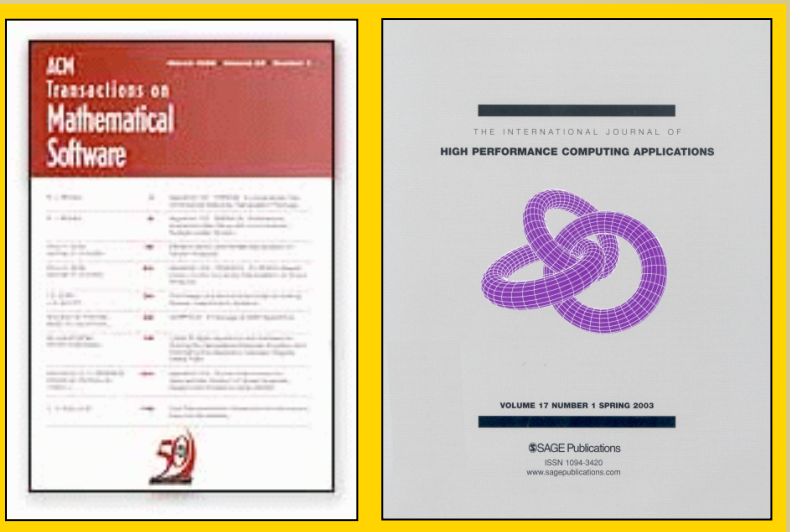
high performance and scalability...

... More Applications …

References

- **An expanded Framework for the Advanced Computational Testing and Simulation Toolkit,** *http://acts.nersc.gov/documents/Proposal.pdf*
- **The Advanced Computational Testing and Simulation (ACTS) Toolkit. *Technical Report LBNL-50414*.**
- **A First Prototype of PyACTS. *Technical Report LBNL-53849*.**
- **ACTS - A collection of High Performing Tools for Scientific Computing. *Technical Report LBNL-53897*.**
- **The ACTS Collection: Robust and high-performance tools for scientific computing. Guidelines for tool inclusion and retirement. *Technical Report LBNL/PUB-3175*.**
- **An Infrastructure for the creation of High End Scientific and Engineering Software Tools and Applications. *Technical Report LBNL/PUB-3176*.**

To appear: two journals featuring ACTS Tools.



Progress Reports

- **FY 2003-2,** *http://acts.nersc.gov/documents/Report2003-2.pdf*
- **FY 2004-1,** *http://acts.nersc.gov/documents/Report2004-1.pdf*
- **FY 2004-2,** *http://acts.nersc.gov/documents/Report2004-2.pdf*
- **FY 2004-3,** *http://acts.nersc.gov/documents/Report2004-3.pdf*

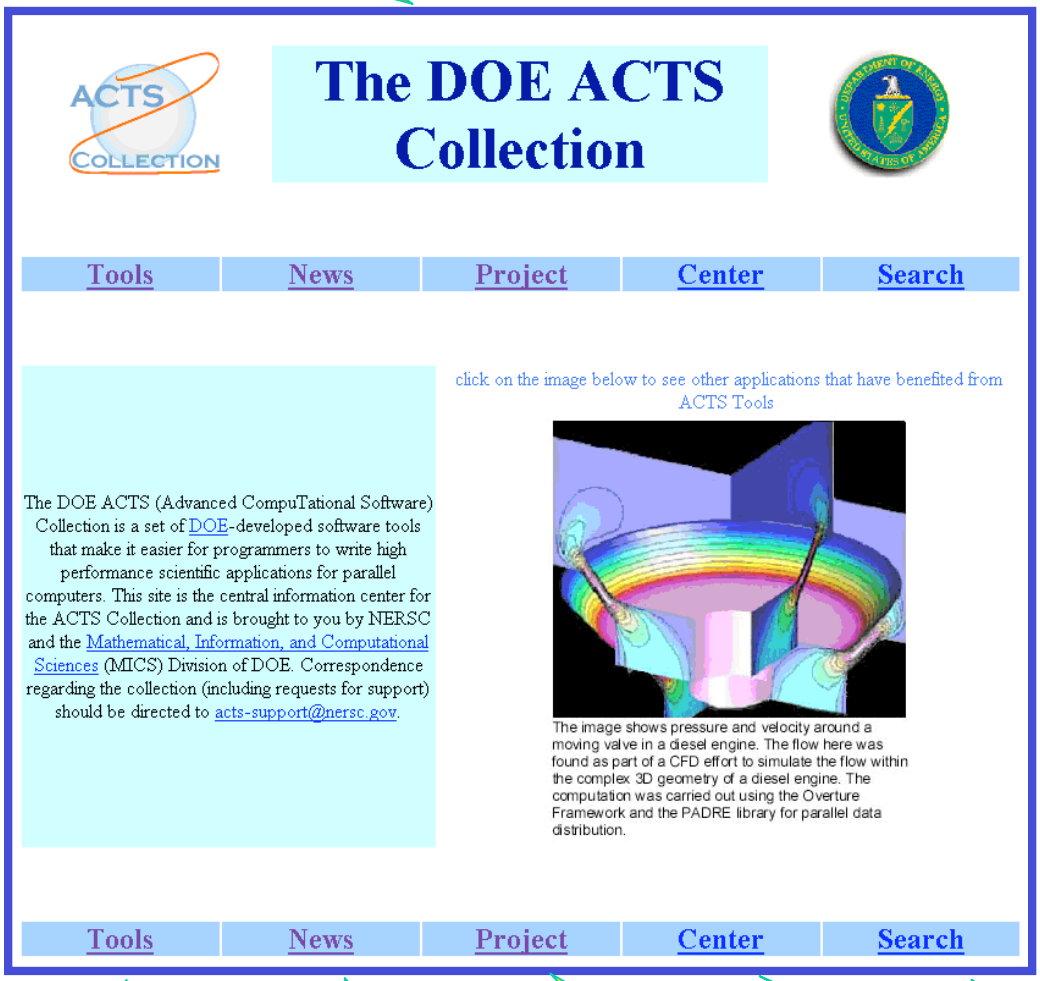
Tutorials and Workshops

- **How Can ACTS Work for you?,** *http://acts.nersc.gov/events/Workshop2000*
- **Solving Problems in Science and Engineering,** *http://acts.nersc.gov/events/Workshop2001*
- **Robust and High Performance Tools for Scientific Computing,** *http://acts.nersc.gov/events/Workshop2002*
- **Robust and High Performance Tools for Scientific Computing,** *http://acts.nersc.gov/events/Workshop2003*
- **The ACTS Collection: Robust and High Performance Libraries for Computational Sciences, SIAM PP04** *http://www.siam.org/meetings/pp04*
- **Enabling Technologies For High End Computer Simulations** *http://acts.nersc.gov/events/Workshop2004*

ACTS COLLECTION

acts-support@nersc.gov
http://acts.nersc.gov

http://acts.nersc.gov



- High Performance Tools
  - portable
  - library calls
  - robust algorithms
  - help code optimization
- Scientific Computing Centers
  - Reduce user's code development time that sums up in more production runs and faster and effective scientific research results
  - Overall better system utilization
  - Facilitate the accumulation and distribution of high performance computing expertise
  - Provide better scientific parameters for procurement and characterization of specific user needs

Tool descriptions, installation details, examples, etc

Agenda, accomplishments, conferences, releases, etc

Goals and other relevant information

Points of contact

Search engine

- **Compframe 2005**
- **ACTS Workshop 2005**